Amendments to the Claims:

This listing of claims will replace all prior versions, and listings of claims in the application:

Listing of Claims:

Claim 1. (Currently Amended) A method for detecting a boundary between two bytes X1 and X2 in a deserialized data stream, the data stream comprising N consecutive X1 bytes followed by N consecutive X2 bytes, the method comprising the steps of:

storing a first M bytes of data, where M is smaller than N;

monitoring at least a subsequent second M bytes of data, wherein the first M bytes of data and the second M bytes of data are from sequential segments of the deserialized data stream;

comparing each of said first M bytes to a value X1*;

comparing each of said second M bytes to a value X2*;

wherein X1* represents X1 or any value resulting from a bit shift of X1,

wherein X2* represents X2 or any value resulting from a bit shift of X2,

and wherein the X1X2 boundary is detected when each of said first M bytes equals X1*, and each of said second M bytes equals X2*.

Claim 2. (Currently Amended) The method of claim ~~2~~ 1 wherein said first M bytes are stored in a first data register.

Claim 3. (Original) The method of claim 2 wherein said data register is a 128 bit register.

Claim 4. (Original) The method of claim 1 wherein the data stream is a portion of a SONET frame.

Claim 5. (Original) The method of claim 4 wherein X1 is the named byte A1 in a SONET frame section header.

Claim 6. (Original) The method of claim 5 wherein X2 is the named byte A2 in a SONET frame section header.

Claim 7. (Original) The method of claim 4 wherein the SONET frame is an OC-N SONET frame, and wherein N represents the number of OC-1 frames multiplexed to form the OC-N frame.

Claim 8. (Original) The method of claim 1 wherein M is substantially equal to half of N.

Claim 9. (Original) A method of aligning data on a data bus along a boundary between two bytes X1 and X2, the values on the data bus resulting from writing consecutive groups of N bytes from a serial data stream, the serial data stream comprising at least N consecutive X1 bytes followed by at least N consecutive X2 bytes, the method comprising the steps of:

storing at least a first set of bus values in a first register;

monitoring the subsequent second set of bus values;

comparing at least the first M bytes of the first register to the value X1*, where X1* represents X1 or any value resulting from a bit shift thereof;

comparing at least the last M bytes of the subsequent set of bus values to a value X2*, where X2* represents X2 or any value resulting from a bit shift thereof;

if each of the first M bytes equals X1* and each of the second M bytes equals X2*, determining the extent to which X1 and X2 are respectively bit shifted from X1* and X2*, and

based on the extent of the bit shift, shifting the bus data in a third register such that mapping the data bus into a second register such that the extent of the bit shift is compensated.

Claim 10. (Original) The method of claim 9 wherein the data bus is a 128 bit wide data bus.

Claim 11. (Original) The method of claim 10 wherein the data registers are 128 bit wide registers.

Claim 12. (Original) The method of claim 9 wherein the data stream is at least some portion of a SONET frame.

Claim 13. (Original) The method of claim 12 wherein X1 is the named byte A1 in a SONET frame section header.

Claim 14. (Original) The method of claim 12 wherein X2 is the named byte A2 in a SONET frame section header.

Claim 15. (Original) The method of claim 12 wherein the SONET frame is an OC-N SONET frame, and wherein N represents the number of OC-1 frames multiplexed to form the OC-N frame.

Claim 16. (Original) The method of claim 9 wherein M is substantially equal to half of N.

Claim 17. (Original) The method of claim 9 further comprising the steps of:

determining the extent to which the data bus is byte shifted with respect to the X1X2 boundary,

and mapping the data bus into a third register such that the third register contains either all X1 values or all X2 values at any point in time.

Claim 18. (Currently Amended) A SONET data processor comprising:

a first register coupled to an input SONET data bus;

a comparator having at least a first input coupled to the input data bus and a second input coupled to the first register such that the comparator has substantially simultaneous access to paralleled data associated with two successive clock cycles,

wherein the comparator compares the values in some portion of the input data bus with a predetermined value, ~~and~~

wherein the comparator compares the values in some portion of the first register with a predetermined value, and

wherein the values in some portion of the input data bus and the values in some portion of the first register are from sequential segments of a deserialized data stream.

Claim 19. (Original) The SONET data processor of claim 18 further comprising:

byte select bus outputted by said comparator whose value is determined by the difference between the values in some portion of the input data bus and a predetermined value, and the difference between some portion of the first register with a predetermined value.

Claim 20. (Original) The processor of claim 19 further comprising:

a second data register coupled to said first data register wherein the second data register stores the values stored in the first data register during a prior clock cycle.

Claim 21. (Original) The processor of claim 20 further comprising:

a bit shifting circuit having at least three inputs and one output, the first input coupled to receive some portion of the first data register's output, the second input coupled to receive some portion of the second data register's output, the third input coupled to receive the bit select bus, and the output coupled to generate a new data comprising bit shifted data wherein each of the bytes in the new data has a value equal to a predetermined value.

Claim 22. (Original) The processor of claim 21 wherein the bit shifting circuit comprises an array of multiplexers.

Claim 23. (Original) The processor of claim 21 wherein the bit shifting circuit comprises an array of multiplexers.

Claim 24. (Original) The processor of claim 23 further comprising a fourth data register having at least one input coupled to the output of the third data register,

wherein the fourth data register stores the values stored in the third data register during a prior clock cycle.

Claim 25. (Original) The processor of claim 24 further comprising byte shifting logic having at least one input coupled to the output of the third data register, and adapted to determine the difference between the value of the third data register and a predetermined value, and to output a byte select control signal whose value is determined in accordance with said difference.

Claim 26. (Original) The processor of claim 25 further comprising a byte shifting circuit having a first input coupled to receive the value of the third data register, a second input coupled to receive the value of the fourth data register, a the third input coupled to receive the byte select control signal, and an output coupled to generate a new data whose value for at least one clock cycle is equal to a predetermined value.

Claim 27. (Original) The processor of claim 26 wherein the byte shifting circuit comprises an array of multiplexers.

Claim 28. (Original) The processor of claim 27 further comprising an output data register having at least one input coupled to the output of the byte shifting circuit.

Claim 29. (Original) A SONET line card comprising:

an optical transceiver coupled to receive an optical signal and to convert the optical signal to an electrical signal;

an electrical transceiver coupled to receive the electrical signal and to deserialize the electrical signal into a plurality of parallel data streams;

a framer coupled to the electrical transceiver and configured to detect an A1A2 boundary of the electrical signal; and

a network processing unit coupled to the framer,

wherein, the framer comprises the SONET data processor of claim 18.

Claim 30. (Currently Amended) A method of processing data in a SONET frame, the method comprising:

receiving first and second consecutive N bytes of data, wherein the first N bytes of data and the second N bytes of data are from sequential segments of a deserialized data stream;

comparing N/2 consecutive bytes of the first N bytes of data with a first predetermined pattern defined by the A1 byte in a SONET frame header;

comparing N/2 consecutive bytes of the second N bytes of data with a second predetermined pattern defined by the A2 byte in a SONET frame header;

if a match is found in both compare steps, forming a third consecutive N+1 bytes by combining the two N/2 consecutive bytes of data plus one additional byte;

shifting data bits in each byte of the third consecutive N+1 bytes so that each byte corresponds to an A1 or an A2 byte; and

shifting the A1 and A2 bytes to align N consecutive bytes along the A1A2 boundary.

Claim 31. (Original) The method of claim 30 wherein the N/2 consecutive bytes of the first N bytes comprises the first half of the first N bytes, and the N/2 consecutive bytes of the second N bytes comprises the second half of the second N bytes.

Claim 32. (Original) The method of claim 30 wherein the N/2 consecutive bytes of the first N bytes comprises the second half of the first N bytes, and the N/2 consecutive bytes of the second N bytes comprises the first half of the second N bytes.

Claim 33. (Original) The method of claim 30 wherein the first predetermined pattern comprises the A1 pattern or any bit shifted version thereof.

Claim 34. (Original) The method of claim 33 wherein the second predetermined pattern comprises the A2 pattern or any bit shifted version thereof.